OPEN ACCESS

KING'S College LONDON

King's Research Portal

DOI:
10.1109/ICDIM.2015.7381869

Document Version
Peer reviewed version

Link to publication record in King's Research Portal

Citation for published version (APA):
Al Khater, N., & Overill, R. E. (2016). Network Traffic Classification Techniques and Challenges. In *The 10th International Conference on Digital Information Management, ICDIM 2015* (pp. 43-48). [7381869] Institute of Electrical and Electronics Engineers Inc.. https://doi.org/10.1109/ICDIM.2015.7381869

Citing this paper
Please note that where the full-text provided on King's Research Portal is the Author Accepted Manuscript or Post-Print version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version for pagination, volume/issue, and date of publication details. And where the final published version is provided on the Research Portal, if citing you are again advised to check the publisher's website for any subsequent corrections.

General rights
Copyright and moral rights for the publications made accessible in the Research Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognize and abide by the legal requirements associated with these rights.

•Users may download and print one copy of any publication from the Research Portal for the purpose of private study or research.
•You may not further distribute the material or use it for any profit-making activity or commercial gain
•You may freely distribute the URL identifying the publication in the Research Portal

Take down policy
If you believe that this document breaches copyright please contact librarypure@kcl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. May. 2023

Network Traffic Classification Techniques and Challenges

Noora Al Khater
Department of Informatics
King's College London
London, United Kingdom
noora.al_khater@kcl.ac.uk

Richard E Overill
Department of Informatics
King's College London
London, United Kingdom
richard.overill@kcl.ac.uk

Abstract—**The number of alleged crimes in computer networks had not increased until a few years ago. Real-time analysis has become essential to detect any suspicious activities. Network classification is the first step of network traffic analysis, and it is the core element of network intrusion detection systems (IDS). Although the techniques of classification have improved and their accuracy has been enhanced, the growing trend of encryption and the insistence of application developers to create new ways to avoid applications being filtered and detected are among the reasons that this field remains open for further research. This paper discusses how researchers apply Machine Learning (ML) algorithms in several classification techniques, utilising the statistical properties of the network traffic flow. It also outlines the next stage of our research, which involves investigating different classification techniques (supervised, semi-supervised, and unsupervised) that use ML algorithms to cope with real-world network traffic.**

Keywords- ***network traffic analysis; Machine Learning (ML); traffic classification; security***

I. INTRODUCTION

Classifying network traffic links network traffic with a generated application, and is a vital first step for network analysis. Valuable information can be gathered from traffic analysis, especially for security purposes such as filtering traffic and identifying and detecting malicious activity. By knowing what type of application is flowing over their networks, network operators can react quickly to potential incidents based on their incident response plans.

Several network traffic classification techniques have been developed over the last two decades to cope with the challenges that classifiers face. Historical developments have revealed significant inaccuracy and unreliability of the traditional techniques (port-based classifications) [1, 2], which depend on port numbers to categorise network traffic. This is because the number of applications that flow over networks using random or non-standard ports has increased exponentially. To overcome this problem, payload-based classification emerged, and inspects not just the headers of the packets but also their contents [2, 3, 4, 5, 6]. This classification is considered a reliable technique with accurate results, but the effectiveness of Deep Packet Inspection (DPI) methods is diminishing since they do not detect traffic encryption and protocol encapsulation.

For these reasons, researchers began to use statistical and behavioural classification applying Machine Learning (ML) techniques. These techniques are used to allocate, control, and manage network resources. Classification is an essential component of intrusion detection systems (IDS), which are used to detect malicious activities. In fact, real-time traffic analysis has the potential to solve problems in network management, security, and forensics by correlating network traffic patterns with the generating applications.

More recently, some governments have required Internet Service Providers (ISPs) to recognise exactly what type of traffic flows over their networks, and have imposed clear ISP obligations to perform lawful interceptions of network traffic [7]. Network traffic classification is thus an essential task to prevent, detect, respond to, and mitigate new forms of attack that can threaten legitimate services and cost organisations a lot.

This paper is structured as follows. Section II demonstrates the development of different classification techniques and discusses their limitations. Section III focuses on the use of machine learning (ML) algorithms in IP traffic, discusses the challenges in this field, and outlines the research gaps. Section IV illustrates the proposed model for network traffic classification. Finally, Section V presents the conclusions and highlights our future work.

II. NETWORK TRAFFIC CLASSIFICATION

Network traffic classification has generated great interest in the research community along with the industrial field. Several techniques have been proposed and developed over the last two decades. This section discusses classification methods and divides them into four categories based on their chronological evolution.

A. Port-based classification

In the early days of the internet, classification and identification of network traffic was not an issue at all. Port-based classification involved identifying an application based on inspecting the packet header and matching it with the TCP or UDP port number registered with the Internet

978-1-4673-9152-8/15/$31.00 ©2015 IEEE

Assigned Numbers Authority (IANA). Unfortunately, historical developments have revealed the inaccuracy and unreliability of these traditional techniques [1, 2]. The diminishing of this technique comes from several causes. Firstly, some modern applications flow with unregistered port numbers (non-standard ports) or pick a random port, for example peer to peer (P2P) applications [8]; in this situation the false negative results of classifier increase. Even worse, other applications (e.g., non-legitimate applications) hide themselves behind well- known ports to avoid being filtered and consequently bypassing restrictions of operation system access control; this raises the false positive rate of classifiers because of undetectable applications. In some situations it is impossible to know the actual port numbers, for example, in the case of obfuscation of the TCP or UDP header by IP layer encryption.

B. Payload-based classification

To overcome the deficiency and reliance on port-based classification, many industry products and research works proposed, based on inspection beyond the headers of the packets to contents, a technique called payload-based classification and sometimes called Deep Packet Inspection (DPI). The most used payload-based technique relies on inspecting packet contents and matching them with a deterministic set of stored signatures (pattern). The results of this method of classification are extremely accurate [2]. Payload examination has been widely used in several commercial and open source tools, for instance, in the implementation of the Linux kernel firewall [9]. Also, payload-based classifiers are often used as a primary step in network intrusion detection systems (IDS) for identifying malicious activity in the network [10]. The reliability of this method has been investigated widely. Sen et al. [6] illustrated that using payload-based classification to identify P2P application traffic could minimise the false positive and false negative rate to 5% in most studied cases. There are several other works based on payload classification [2, 3, 4, 5].

Although payload-based examination is considered a reliable technique, it has some significant disadvantages and weaknesses. First, the ability of the classifier diminishes when dealing with encrypted traffic and protocol obfuscation or encapsulation; examining an encrypted packet with this method is impossible, which means a lot of network traffic remains unclassified. Besides this and more importantly, inspecting contents of a packet poses privacy challenges and this act may represent a breach of privacy policies and regulations. Furthermore, this method imposes a high computational cost and load on the classification device, as it requires several instances of access to the packet contents. Consequently, payload-based classification faces difficulty in coping with the large number of flows and high speed rate of network traffic.

C. Statistical classification

Statistical classification is a rationale-based technique that exploits statistical characteristics of network traffic flow to identify the application. This method utilises a number of flow-level measurements [11, 12, 8], for example, the duration of the packet, packet inter-arrival time, packet lengths, and traffic flow idle time. These measurements are unique for specific type of applications; hence, this allows the classifier to distinguish different applications from each other.

In the early stage, the statistical characteristics of network traffic were investigated in several studies. In [13] the authors observed the correlation between the class in network traffic and the statistical characteristics of the flow such as bytes and duration. They proposed experiential models of connection characteristics for a large number of TCP applications. Also, in [14] the authors utilised the statistical characteristics of the network flow, such as packet size, packet inter-arrival time, and the flow duration, to analyse internet chat systems. Some of the later work such as [15], [16] and [17] observed the unique properties of network traffic for a number of internet applications. The outcomes of these studies have inspired researchers to work on new classification techniques based on statistical properties.

To perform the actual classification based on statistical characteristics, classifiers need to employ data mining techniques, specifically ML algorithms, because they need to deal with different traffic patterns from large datasets. ML algorithms are very lightweight and less computationally expensive than payload-based classification techniques, because they do not depend on DPI but rather utilise the information from flow-level analysis. The effectiveness of the classifier in statistical classification depends on the features extracted from the flow, which require extensive knowledge due to their complexity. However, these techniques outperform payload-based techniques since they do not deal with packet contents, and thus can analyse encrypted traffic without any difficulty.

D. Behavioral classification

The behavioral classification technique observes the whole network traffic received by the endpoint (host), seeking to identify the type of application by analysing the generated network traffic patterns from the target host. For example, the number of communicated hosts is counted, taking into account the transport layer protocol and the number of ports.

Some researchers such as [18, 19] sought to analyse network traffic patterns by exploiting heuristic information such as the number of distinct ports contacted, as well as transport layer protocols to distinguish the type of application running on a host. Other works [20, 21] showed that a lot of information can be utilised to classify network traffic. They analyzed the connections between endpoints graphically, and they show that generated connection patterns and graphs from client-server applications are very different than those of P2P. Another group of researchers [22, 23] exploited the power of ML algorithms and combined them with metrics to classify specific applications in the network. Although the behavioural classification technique yields promising results with lower computational

cost, most of these proposed works studied only the end-hosts [24] or endpoint [19] activity. The limitations of this research are inherent to the methodology they used.

III. Challenges in Network traffic Classification Using ML techniques

In fact, over the last few years, most internet applications have used a well-known port over a transport layer protocol, which makes them easily and precisely identified. However, nowadays the classification task is further exacerbated and the mission has become harder for several reasons:

- The classifier analysers must deal with the increasing amount of traffic and transmission rates;
- Researchers are looking for lightweight algorithms with little computational cost;
- The growing trend of traffic encryption and protocol encapsulation in the network poses further challenges; and
- Application developers continue to invent new ways to prevent traffic being filtered and detected.

These reasons were the motivation for researchers to move toward applying ML techniques to classify network traffic based on statistical and behavioural features.

In fact, the majority of ML techniques which are used for IP traffic classification focus on the use of supervised and unsupervised learning (clustering), while a few use hybrid techniques (semi-supervised). Roughan et al. [8] applied ML algorithms, Nearest Neighbours (NN), Linear Discriminate Analysis (LDA) and Quadratic Discriminant Analysis (QDA) to classify IP traffic based on the statistical signature approach. Classification results show that three-class classification has the minimum error rate. The error rate increases when more applications are mixed, which explains the highest error rate in seven-class classification. This means the efficiency of the classifier decreases when it deals with different applications.

Moore and Zuev [12] studied how to categorise network traffic by application using the supervised ML Naive Bayes technique. This study was amended and improved by [25], by using the Bayesian neural network method. More accurate results were achieved compared to the previous work.

One of the earliest study such as [11] used unsupervised techniques by applying the Expectation Maximisation (EM) algorithm [26] to cluster the traffic with similar characteristics into different application groupings. The collected features are based on full flow. The EM algorithm is applied to cluster the network traffic into a number of groups and create classifier rules based on the clusters. The useless and ineffective features are discarded and removed from the input and the learning course is repeated. Although the classification results are bounded by identifying particular applications, this method could be used as a first step to classify unknown traffic to give a clue about the application groups in traffic.

The proposed approach by Zander et al. [27] used ML techniques based on statistical flow properties and utilised the unsupervised Bayesian classifier AutoClass [28] for application identification. The authors applied the EM algorithm to identify the most proper set from training data. AutoClass can calculate approximately the number of classes, if not configured previously. Their technique is also based on full flow to calculate features.

Other published studies such as [29] and [30] also aimed to investigate the performance of ML, but by exploiting the first few packets. Although this technique is considered to be faster and less time consuming than exploitation on a full flow basis, the ability of this classifier deteriorates if the initial packets are lost.

Crotti et al. [31] proposed the protocol fingerprint method, and classified network traffic applying an algorithm based on normalised thresholds. The proposed technique relied on three characteristics of the collected IP packets (inter-arrival time and arrival order of the packets as well as their length). Their study results achieved high accuracy for identifying three kinds of applications using the first few packets as [29]. Nevertheless, the effectiveness of the method deteriorates if the classifier is not aware of the locations of the client and server, if the beginning of the flow is missed, if the first packet is lost or if packet reordering is not included.

Indeed, most of the studies explored the effectiveness of ML algorithms by training and testing classifiers over full flows [8, 11, 12, 27, 30, 32], whereas identifying the type of application before the flow ends can avoid a lot of losses for organisations in case of security events. A few research studies (e.g. [33, 34]) evaluated ML techniques using a sub-flow. The authors in [33] proposed a classification method based on sub-flows instead of relying on classification based on features extracted from full flows. They applied the Naive Bayes ML algorithm with using a small amount of the most recent packets extracted from full flow to train the classifier, they proved that they were able to minimise buffer space in classification process. Furthermore, this technique avoided the classifier's search for the start of the flow (like the studies [29] and [4]), which might be missed and consequently affect performance of the classifier; this demonstrated poor performance of full flow-based classification in some scenarios. This study was extended in [35], by applying Naive Bayes and decision tree algorithms, with using the same datasets as in the previous work.

It is noticeable that most of the research so far has not evaluated the performance of the classifier in terms of packet loss, fragmentation, and delay. Moreover, although unsupervised techniques have the ability to detect the emergence of a new application in the network, the majority of the works have not investigated this issue, even though it is mentioned in [36]. The authors in [36] combined two ML algorithms, unsupervised and supervised, their approch takes advantages of both techniques. Supervised methods classify flow based on former experience, using examples from training phase, while unsupervised classification methods

have the ability to detect new types of applications without any guidance. The proposed method has more advantages than classification using only supervised ML algorithms. This technique reduces the time needed to train the classifier by using a small amount of labelled flows. It is able to handle degradation of supervised algorithms when dealing with unseen examples by employing clustering techniques, which enhances the classifier's performance. However, these benefits of using semi-supervised techniques were not evaluated in this study.

IV. NETWORK TRAFFIC CLASSIFICATION MODEL

This section illustrates the proposed structure and processes involved in network traffic classification as shown in Fig. 1. This demonstrates the sequential steps that we will use in our research project to analyse network traffic by correlating network traffic patterns with the generating applications using ML techniques.

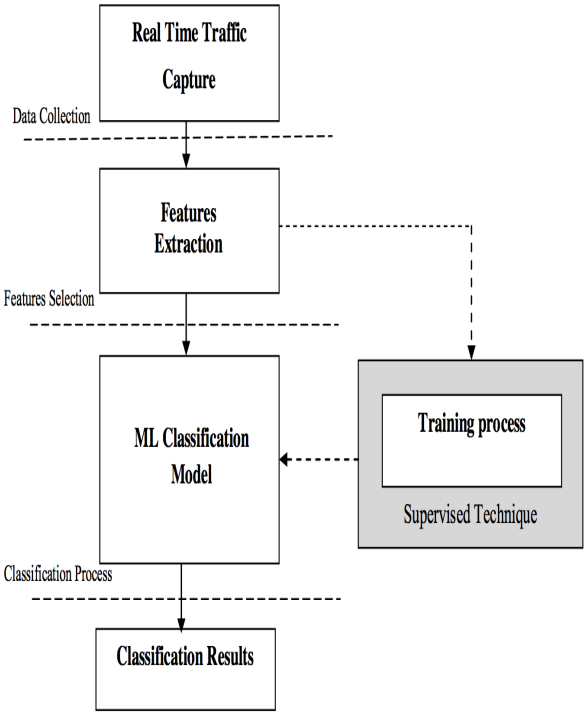

Figure 1 Network traffic classification model

A. Data collection

This is the first step, dealing with capturing real-time traffic from networks. Subsequently, the collected dataset is used for feature extraction, to train and test classifiers.

B. Features selection

This step deals with extracting features from network flow. Inter-packet arrival times, packet length, and packet duration are examples of features that will be used to train the classifier. This is the core element to build a robust classifier. In this research will be based on a sub-flow instead of the full flow or the first few packets in the flow. This avoids having to wait for a full flow and minimises the required space for a buffer, which consequently increases the memory space. This aspect has not yet been explored by most of the published researches.

C. Training process

The training phase involves data sampling and classifier training based on the (pre-labelled) samples. This step uses supervised algorithms to create classification rules and build the model. The information learnt during the training phase is used to classify new unseen examples in the testing phase (classification process).

D. Classification process

This is the final stage, where network traffic analysis accrues, by correlating network traffic patterns with the generating applications. In this stage our research project will examine different classification techniques as shown in Fig. 2. supervised, semi-supervised and unsupervised ML algorithms based on statistical analysis of sub-flows. By doing so we are aiming to explore the extent of their robustness and effectiveness in classifying real- world traffic correctly and to investigate their ability to detect new emerging applications, which could be malicious applications.

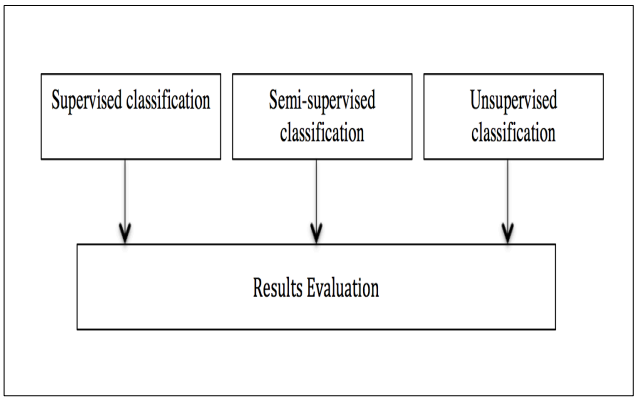

Figure 2 Workflow of different ML traffic classifiers

V. CONCLUSION

The increased number of security threats and crimes conducted in cyberspace proves that there is a substantial amount of network traffic that is still unclassified, along with unauthorised access that passes all the security systems and regulations without any detection.

This paper provides a critical review of the field of network traffic analysis, and focuses on the use of ML algorithms to classify internet traffic. It demonstrates the great interest of the researchers in this topic over all the stages of IP classification, besides highlighting the issues associated with classification methodologies that have been

used profusely by researchers. It is clear that ML algorithms can be utilised very well in this area. However, this study shows that the majority of ML techniques which are used for IP traffic classification focus on the use of supervised and unsupervised learning (clustering), while a few use hybrid techniques (semi-supervised). Moreover, most of the proposed works are based on statistical features extracted from full flows or just the first few packets in the flows, while a few research works have explored the use of sub-flows where using sub-flows seems to be the most appropriate approach for faster recognition and timely detection. Therefore, the next stage of this research will investigate different classification techniques using ML algorithms based on statistical features extracted from sub-flows. In this work, we are seeking to identify ways to improve classification methods, which aim to develop more powerful and effective techniques to correctly classify network traffic.

REFERENCES

[1] T. Karagiannis, A. Broido, N. Brownlee, K. C. Klaffy, and M. Faloutsos, "Is P2P dying or just hiding?" In IEEE Global Telecommun. Conf. 2004, Dallas, TX, vol. 3, pp. 1532-1538, 2004.

[2] A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," In Passive and Active Measurement 2005, Boston, MA, pp. 41-54, March 2005.

[3] A. Finamore, M. Mellia, M. Meo, and D. Rossi, "Kiss: Stochastic packet inspection classifier for UDP traffic," IEEE/ACM Trans. Netw., vol. 18, no. 5, pp. 1505– 1515, 2010.

[4] P. Haffner, S. Sen, O. Spatscheck, and D. Wang, "ACAS: automated construction of application signatures," in MineNet 2005: Proc. 2005 ACM Special Interest Group on Data Communication Workshop on Mining Network Data, Philadelphia, PA, August 2005, pp. 197–202.

[5] J. Ma, K. Levchenko, C. Kreibich, S. Savage, and G. M. Voelker, "Unexpected means of protocol inference," in 6th ACM Special Interest Group on Data Communication Workshop Internet Measurement Conf. 2006, Rio de Janeiro, pp. 313-326, October 2006.

[6] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in-network identification of P2P traffic using application signatures," in 13th Int. Conf. World Wide Web 2004, New York, NY, pp. 512-521, May 2004.

[7] F. Baker, B. Foster, and C. Sharp, "Cisco architecture for lawful intercept in IP networks," Internet Engineering Task Force, RFC 3924, 2004.

[8] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "Class-of-service mapping for QoS: A statistical signature-based approach to IP traffic classification," in Proc. ACM/ Special Interest Group on Data Communication Internet Measurement Conf. 2004, Taormina, Sicily, pp. 135-148, October 2004.

[9] J. Levandoski, E. Sommer, and M. Strait, , "Application layer packet classifier for Linux", 2008.

[10] V. Paxson, "Bro: a system for detecting network intruders in real-time," Elsevier Comput. Netw., vol. 31, pp. 2435–2463, December 1999.

[11] A. Mcgregor, M. Hall, P. Lorier, and J. Brunskill, "Flow clustering using machine learning techniques," in Proc. Passive and Active Measurement Workshop 2004, Antibes Juan-les-Pins, France, pp. 205-214 , April 2004.

[12] A. Moore and D. Zuev, "Internet traffic classification using Bayesian analysis techniques," in ACM Int. Conf. Measurement and Modeling of Computer Systems (SIGMETRICS) 2005, Banff, Alberta, Canada, pp. 50-60, June 2005.

[13] V. Paxson, "Empirically derived analytic models of wide-area TCP connections," IEEE/ACM Trans. Netw., vol. 2, no. 4, pp. 316–336, 1994.

[14] C. Dewes, A. Wichmann, and A. Feldmann, "An analysis of Internet chat systems," in ACM/SIGCOMM Internet Measurement Conf. 2003, Miami, FL, pp. 51-64, October 2003.

[15] K. Claffy, "Internet traffic characterisation," PhD dissertation, Univ. California, San Diego, CA, Jun1994.

[16] T. Lang, G. Armitage, P. Branch, and H.-Y. Choo, "A synthetic traffic model for half-life," in Proc. Australian Telecommunications Networks and Applications Conf. 2003, Melbourne, Australia, December 2003.

[17] T. Lang, P. Branch, and G. Armitage, "A synthetic traffic model for Quake 3," in Proc. ACM SIGCHI Int. Conf. Advances in Computer Entertainment Technology 2004, Singapore, pp. 233-238, June 2004.

[18] T. Karagiannis, A. Broido, M. Faloutsos, and K. C. Claffy, "Transport layer identification of P2P traffic," in 4th ACM Special Interest Group on Data Communication Internet Measurement Conf. 2004, Taormina, Italy, pp. 121-134, October 2004.

[19] K. Xu, Z.-L. Zhang, and S. Bhattacharyya, "Profiling Internet backbone traffic: Behavior models and applications," ACM SIGCOMM Comput. Commun. Rev., vol. 35, no. 4, pp. 169–180, 2005.

[20] M. Iliofotou, P. Pappu, M. Faloutsos, M. Mitzenmacher, S. Singh, and G. Varghese, "Network monitoring using traffic dispersion graphs (TDGs)," in Proc. Internet Measurement Conf. 2007, San Diego, CA, pp. 315-320, 2007.

[21] Y. Jin, N. Duffield, P. Haffner, S. Sen, and Z.-L. Zhang, "Inferring applications at the network layer using collective traffic statistics," SIGMETRICS Perform. Eval. Rev., vol. 38, p. 1-8, June 2010.

[22] P. Bermolen, M. Mellia, M. Meo, D. Rossi, and S. Valenti, "Abacus: Accurate behavioral classification of P2P-TV traffic," Elsevier Comput. Netw., vol. 55, no. 6, pp. 1394–1411, 2011.

[23] T. Z. J. Fu, Y. Hu, X. Shi, D.-M. Chiu, and J. C. S. Lui, "PBS: Periodic behavioral spectrum of P2P applications," in Proc. Passive and Active Network Measurement 2009, Seoul, South Korea, pp. 155-164 , April 2009.

[24] T. Karagiannis, K. Papagiannaki, N. Taft, and M. Faloutsos, "Profiling the end host," in Proc. 8th Int. Conf. Passive and Active Network Measurement 2007, Louvain-la-Neuve, Belgium, pp. 186-196, 2007.

[25] T. Auld, A. W. Moore, and S. F. Gull, "Bayesian neural networks for Internet traffic classification," IEEE Trans. Neural Netw., vol. 18, no. 1, pp. 223–239, January 2007.

[26] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," J. Roy. Stat. Soc., vol. 30, no. 1, pp. 1-38, 1997.

[27] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," in IEEE 30th Conf. Local Computer Networks 2005, Sydney, Australia, pp. 250-257, November 2005.

[28] P. Cheeseman and J. Stutz, "Bayesian classification (AutoClass): Theory and results," in Advances in Knowledge Discovery and Data Mining, 1996.

[29] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly," ACM SIGCOMM Comput. Commun. Rev., vol. 36, no. 2, pp. 23-26, 2006.

[30] J. Erman, A. Mahanti, and M. Arlitt, "Internet traffic identification using machine learning techniques," in Proc. 49th IEEE Global Telecommunications Conf. 2006, San Francisco, CA, December 2006.

[31] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli, "Traffic classification through simple statistical fingerprinting," SIGCOMM Comput. Commun. Rev., vol. 37, no. 1, pp. 5–16, 2007.

[32] N. Williams, S. Zander, and G. Armitage, "A preliminary performance comparison of five machine learning algorithms for

practical IP traffic flow classification," Special Interest Group on Data Communication (SIGCOMM), vol. 36, no. 2, pp. 5-16, 2006.

[33] T. Nguyen and G. Armitage, "Training on multiple sub-flows to optimise the use of machine learning classifiers in real-world IP networks," in Proc. IEEE 31st Conf. Local Computer Networks, Tampa, FL, pp. 369-376, November 2006.

[34] T. T. T. Nguyen, G. Armitage, P. Branch, and S. Zander, "Timely and continuous machine-learning-based classification for interactive IP traffic," IEEE/ACM Trans. Netw., vol. 20, no. 6, pp. 1880-1894, December 2012.

[35] T. Nguyen and G. Armitage, "Synthetic sub-flow pairs for timely and stable IP traffic identification," in Proc. Australian Telecommunication Networks and Application Conf., Melbourne, Australia, December 2006.

[36] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semisupervised network traffic classification," ACM Int. Conf. Measurement and Modeling of Comput. Syst. (SIGMETRICS) Performance Evaluation Rev., vol. 35, no. 1, pp. 369–370, 2007.